

IN THE UNITED STATES PATENT AND TRADEMARK OFFICE

In Re The Application of:
Vinay Gupta

Serial No.: 10/772,822

Filed: February 5, 2004

For: SYSTEM AND METHOD FOR
LUN CLONING

Examiner: Savla, Arpan P

Art Unit: 2185

Confirmation No.: 5952

Cesari and McKenna, LLP
88 Black Falcon Avenue
Boston, MA 02210
May 7, 2009

CERTIFICATE OF TRANSMISSION

I hereby certify that the following paper(s) is/are being electronically transmitted
to the Patent and Trademark Office by EFS-Web on May 7, 2009

/Stephany Elmore/
Stephany Elmore

Commissioner for Patents
P.O. Box 1450
Alexandria, VA 22313-1450

Sir:

DECLARATION OF PRIOR INVENTION

TO OVERCOME CITED PUBLICATION UNDER 37 C.F.R. § 1.131

This document is provided as a declaration of prior invention to overcome cited
publications under 37 C.F.R. § 1.131.

PURPOSE OF DECLARATION

1. This declaration is to establish completion of the invention of this application in The United States at a date prior to the effective date of the prior art publications, Federwisch et al., U.S. Publication No. 2003/0182313 published September 25, 2003, and Edwards, U.S. Publication No. 2003/0182389 published September 25, 2003, both of which were cited by the Examiner.
2. The person making this declaration is Vinay Gupta, a joint inventor.

FACTS AND DOCUMENTARY EVIDENCE

3. To establish the date of completion of the invention of this application, the following attached document is submitted as evidence:
“Lazy Lun Cloning Specification”.

4. Mr. Vinay Gupta, a joint inventor, swears that the conception of the invention was prior to September 25, 2003, the effective dates of the references.

DILIGENCE

5. Mr. Vinay Gupta acknowledges through this declaration that Applicant acted with diligence in the completion of the invention from the time of conception, to a time just prior to the date of the reference, up to the filing of this application.

DECLARATION

6. I, Vinay Gupta, as a joint inventor of the present invention, hereby declare that all statements made herein of my own knowledge are true and that all statements made on information and belief are believed to be true; and further that these statements were made with the knowledge that willful false statements and the like so made are punishable by fine or imprisonment, or both, under Section 1001 of Title 18 of the United

States Code, and that such willful false statements may jeopardize the validity of the application or any patent issued thereon.

SIGNATURE

7. Full name of (joint/sole) inventor (please print): VINAY GUPTA

Inventor's signature: Vinay Gupta Date: 04/29/09

Country of Citizenship: INDIA

Country of Residence: US

Post Office Address: 140 E 14th Street, #1534A
New York, NY 10003

Notarized by: Debra A. Cunningham

Signature of Notary: Debra A. Cunningham Date: 4/29/09

My Commission Expires: 11-16-2010

SEAL



Please charge any additional fee occasioned by this paper to our Deposit Account

No. 03-1237.

PATENTS
112056-0159
P01-1727

Respectfully submitted,



A. Sidney Johnston
Reg. No. 29,548
CESARI AND MCKENNA, LLP
88 Black Falcon Avenue
Boston, MA 02210-2414
(617) 951-2500

Lazy LUN Cloning Specification

Introduction

Lazy LUN cloning allows users to clone existing LUNs with zero-downtime. LUNs backed by a snapshot have part of their data blocks in a snapshot and part of it in the active file system. Thus as long as the LUN is being used, the backing snapshot cannot be deleted, which locks in the blocks from other files on the volume as well.

It is desired that it should be possible to delete the snapshot after some reasonable period of time without loosing the contents of the LUN and without taking the LUN offline.

The proposed LUN cloning technique allows this by cloning the contents of the LUN in the background, thus "Lazy", without any application visible downtime. Once the cloning is complete the backing snapshot can then be deleted.

Note that SFSR is not a solution here since we wish to retain the contents of the LUN as is and not restore to the version in the snapshot. Further for SFSR the LUN would be unavailable for the duration of the operation and thus not a solution to the problem at hand.

Design Details

In the initial cut of the implementation, the cloning will be done entirely using backdoor messages and no nvlog protection.

It is not necessary to avoid the buffer copy when a level-0 buffer does not exist in the active filesystem and thus forms the basis of our implementation.

`wafl_load_buf()` on the active filesystem inode already fetches the correct contents, from the most recent applicable snapshot file. This is true for level-0 blocks. For levels greater than 0, the contents are those corresponding to the active filesystem only.

`vdisk_backup_compute_level_one_buf()` computes a level-1 buffer. (When writable snapshot files have holes, this loads all backing inodes, and composes the result buffer.)

We compare the level-1 buffers from the inode in the active filesystem and from that in the snapshot and if there are any holes in the active filesystem we load the corresponding level-0 buffers and mark them dirty.

If a buffer for the active filesystem inode is marked dirty, it is written back to the active filesystem inode, in the normal manner.

The cloning process is complete when the backdoor message handler returns successfully. At this point the lock on the snapshot is released, the entry in the vtoc is updated to remove the snapshot reference and the user informed of the completion through a syslogged message.

- The cloning is done entirely in the background without any NVLOG protection.
• If we panic in the middle of a cloning operation, the operation is restarted from the beginning on a reboot/takeover (see Future below).

Implementation Details

- + load writable snapshot vdisk's level-1 buffers using `wafl_load_buf`
- + load corresponding level-1 buffers for backing snapshot file, with `vdisk_backup_compute_level_one_buf()`, instead.
- + where level-0 vbn for active filesystem inode is 0, and for the backing file is non-zero:
 - load the active filesystem inode level 0 buffer
 - `wafl_set_buf_dirty()` on the buffer
- + If a complete scan is performed, and no buffers were marked dirty, do a `backdoor_send` operation to update the vtoc entry
- + Otherwise do `wafl_sync()` to write the dirty buffers to disk and then update the vtoc entry

Other UI Details

- A new lun subcommand "lun clone" will be introduced with following sub options:
 - + "lun clone show [<path>]" will list all the luns being cloned
 - + "lun clone status <path>" will report the progress on the cloning operation
 - + "lun clone start <path>" will start of the cloning process
 - + "lun clone stop <path>" will stop the operation

Future

1. Make multiple clones of a single lun simultaneously
2. Keep track of the amount of work done and avoid redoing all the work on a reboot.

Test Plan

TBD

Tracking Burt

burt 77100 RFE: Simple LUN cloning implementation
(Most of the text in this specification is borrowed from Vijayan's Notes in this burt.)

vinay@netapp.com